INRIA

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Bandwidth selection for kernel estimation in mixed multi-dimensional spaces

Aurélie Bugeau — Patrick Pérez

N° 6286

September 2007

Thèmes COM et COG

ISSN 0249-6399 ISRN INRIA/RR--6286--FR+ENG

Bandwidth selection for kernel estimation in mixed multi-dimensional spaces

Aurélie Bugeau , Patrick Pérez

Thèmes COM et COG — Systèmes communicants et Systèmes cognitifs
Projet Vista

Rapport de recherche n° 6286 — September 2007 — 26 pages

Abstract: Kernel estimation techniques, such as mean shift, suffer from one major drawback: the kernel bandwidth selection. The bandwidth can be fixed for all the data set or can vary at each points. Automatic bandwidth selection becomes a real challenge in case of multidimensional heterogeneous features. This paper presents a solution to this problem. It is an extension of [4] which was based on the fundamental property of normal distributions regarding the bias of the normalized density gradient. The selection is done iteratively for each type of features, by looking for the stability of local bandwidth estimates across a predefined range of bandwidths. A pseudo balloon mean shift filtering and partitioning are introduced. The validity of the method is demonstrated in the context of color image segmentation based on a 5-dimensional space.

Key-words: kernel estimation, mean shift filtering, automatic bandwidth selection

Unité de recherche INRIA Rennes
IRISA, Campus universitaire de Beaulieu, 35042 Rennes Cedex (France)
Téléphone : +33 2 99 84 71 00 — Télécopie : +33 2 99 84 71 71

Estimation à noyau adaptatif dans des espaces multidimensionnels hétérogènes

Résumé : Les méthodes d'estimation à noyau, telles que le mean shift, ont un inconvénient majeur : le choix de la taille du noyau. La taille peut être fixe pour l'ensemble des données ou varier en chaque point. La sélection de cette taille devient vraiment difficile dans le cas de données multidimensionnelles et hétérogènes. Ce rapport présente une solution à ce problème. Il s'agit d'une extension de l'algorithme présenté dans [4]. La taille est choisie itérativement pour chaque type de données, en cherchant dans un ensemble de tailles prédéfinies celle qui donne localement les résultats les plus stables. La sélection itérative nécessite l'introduction d'un algorithme de segmentation mean shift basé sur l'estimateur dit balloon. La méthode est validée dans le contexte de la segmentation d'image couleur.

Mots-clés : estimateur à noyau, filtrage mean shift, taille de noyau

Contents

1 Introduction

Clustering is an important task in a wide range of applications of computer vision. Many methods exist [11]. Most of them rely upon some *a priori*. For example, for methods such as EM, the number of clusters must be known beforehand. It can be estimated by optimizing a global criterion. Other methods assume known the shape, often elliptical, of the clusters, which is often not sufficient to handle the complexity of real images. A method that does not rely on these two priors is the "mean

shift" search for the modes of a kernel density estimation. The non-parametric aspect of the approach makes it very versatile to analyze arbitrary feature spaces. Hierarchical clustering methods are also non-parametric. However, they are computationally expensive and defining the stopping criterion is not simple. These reasons explain why the mean shift clustering became recently so popular in computer vision applications.

Mean shift was first introduced by Fukunaga [9] and latter by Cheng [3]. It has then been widely studied, in particular by Comaniciu [7, 6, 5]. Mean shift is an iterative gradient ascent method used to locate the density modes of a cloud of points, *i.e.* the local maxima of its density. The estimation of the density is done through a kernel density estimation. The difficulty is to define the size of the kernel, *i.e.* the bandwidth matrix. The value of the bandwidth matrix highly influences the results of the mean shift clustering.

There are two types of bandwidth matrices. The first ones are fixed for the all data set. At the opposite, the variable bandwidth matrices vary along the set and capture the local characteristics of the data. Of course, the second type is more appropriate for real scenes. In fact, a fixed bandwidth affects the estimation performance by undersmoothing the tails of the density and oversmoothing the peaks. A variable bandwidth mean shift procedure has been introduced in [5]. It is based on the sample point density estimator [10]. The estimation bias of this estimator decreases in comparison to the fixed-bandwidth estimators, while the covariance remains the same. The choice of a good value for the bandwidth matrix is really essential for the variable bandwidth mean shift. Indeed, when the bandwidth is not selected properly, the performance is often worse than with a fixed bandwidth. Another variable bandwidth estimator is the balloon estimator. It suffers of several drawbacks and has therefore never been used in a mean shift algorithm. However, it has been shown in [23] that this estimator gives better result than the fixed bandwidth and the sample point estimators when the dimensionality of the data is higher than three. Hence, in section 3.3.2, we will propose a new mean shift clustering algorithm based on the balloon estimator.

The bandwidth selection can be statistical analysis-based or task-oriented. Statistical analysis-based methods compute the best bandwidth by balancing the bias against the variance of the density estimate. Task-oriented methods rely on the stability of the feature space partitioning. For example, a semi parametric bandwidth selection algorithm, well adapted for variable bandwidth mean shift, has been proposed by Comaniciu in [4, 7]. It works as follows. Fixed-bandwidth mean shift partitionings are run on the data for several predefined bandwidth values. Each cluster obtained is described by a normal law. Then, for each point, the clusters to which it belongs across the range of predefined bandwidths are compared. The final selected bandwidth for this point corresponds to the one, within the predefined range, that gave the most stable among these clusters. The results obtained for color segmentation were promising. However, this method has some limits. In particular, in case of a multidimensional data points composed of independent features, the bandwidth for each feature subspace should be chosen independently. Indeed, the most stable cluster is not always the same for all the feature subspaces. A solution could be to define a set of bandwidths for each domain and to partition the

data using all the possible bandwidth matrices resulting from the combination of the different sets. However as the dimensions become high and/or if the sets of predefined bandwidths become large, the algorithm can become very computationally expensive.

In this paper we address the problem of data-driven bandwidth selection for multidimensional data composed of different independent features (a data point is a concatenation of different, possibly multidimensional features, thus living in a product of different feature spaces). As no statistically founded method exists for variable bandwidth and for high dimension, we concentrate on a task-oriented method, *i.e.* a method that relies on the stability of the feature space partitionings. Bandwidths are selected by iteratively applying the stability criteria of [4] for each different feature space or domain. We also introduce a new pseudo balloon mean shift which is better adapted for high dimensional feature spaces than the variable bandwidth mean shift of [7].

We first recall some theory on kernel density estimation (section 2) and mean shift filtering (section 3), and introduce the pseudo balloon mean shift filtering and partitioning (subsection 3.3.1). In section 4, we present our algorithm for bandwidth selection algorithm in case of multivariate data and finally we show results of our algorithm for color clustering and color image segmentation (section 5).

2 Kernel density estimation

For the clarity of the paper, we start by reminding several results on fixed and variable bandwidth kernel density estimation.

2.1 Fixed bandwidth estimator

Given $\{\mathbf{x}^{(i)}\}_{i=1..n}$, n points in the d-dimensional space $\mathbb{R}^d$, the non-parametric kernel density estimation at each point $\mathbf{x}$ is given by:

$$\widehat{f}(\mathbf{x}) = \frac{1}{n}\sum_{i=1}^{n} K_{\mathbf{H}}(\mathbf{x}-\mathbf{x}^{(i)}) \tag{1}$$

where $K_{\mathbf{H}}$ is a kernel, and the bandwidth matrix, $\mathbf{H}$, controls the size of the kernel. The shape of the kernel is constrained to be spherically symmetric. Equation 1 can also be written as:

$$\widehat{f}(\mathbf{x}) = \frac{1}{n|\mathbf{H}|^{1/2}}\sum_{i=1}^{n} K(\mathbf{H}^{-1/2}(\mathbf{x}-\mathbf{x}^{(i)})) \ . \tag{2}$$

The theory of kernel density estimation indicates that the kernel K must be a bounded function with compact support satisfying:

$$\begin{aligned} \int_{\mathbb{R}^d} K(\mathbf{x})\mathrm{d}\mathbf{x} = 1 \ , \quad \lim_{\|\mathbf{x}\|\to\infty} \|\mathbf{x}\|^d K(x) = 0 \ , \\ \int_{\mathbb{R}^d} \mathbf{x}K(\mathbf{x})\mathrm{d}\mathbf{x} = 0 \ , \quad \int_{\mathbb{R}^d} \mathbf{x}\mathbf{x}^T K(\mathbf{x})\mathrm{d}\mathbf{x} = c_k\mathbf{I} \ , \end{aligned} \tag{3}$$

where c_k is a constant and $\mathbf{I}$ is the identity matrix.

In many cases fixed bandwidth kernel estimators are not a good choice to represent the data. Indeed a variable bandwidth is more appropriate to capture the local characteristics of the data. Two main variable bandwidth estimators exist. The first one allows the definition of bandwidths at the different data points and is referred to as the sample point estimator. The second one lets the bandwidth vary with the estimation points and is often referred to as the balloon estimator or nearest neighbor estimator.

2.2 Sample point estimator

The sample point estimator was first introduced by Breiman *et al.* [12]. It is a mixture of similar kernels centered on data points, possibly with different bandwidths. It is defined as:

$$\begin{aligned}\widehat{f}(\mathbf{x}) &= \frac{1}{n}\sum_{i=1}^{n} K_{\mathbf{H}(\mathbf{x}^{(i)})}(\mathbf{x}-\mathbf{x}^{(i)}) \\ &= \frac{1}{n}\sum_{i=1}^{n} \frac{1}{|\mathbf{H}(\mathbf{x}^{(i)})|^{1/2}} K(\mathbf{H}(\mathbf{x}^{(i)})^{-1/2}(\mathbf{x}-\mathbf{x}^{(i)})) \ .\end{aligned} \tag{4}$$

In [23] the advantages and drawbacks of this estimator have been studied. The major advantages are that it is a density and that a particular choice of $\mathbf{H}(x^{(i)})$ can considerably reduce the bias [10]. However finding this value for multivariate data is a hard problem not yet solved. A disadvantage is that the estimate at a point may be influenced by observations very far away and not just by points nearby. In [23] simulations have shown that this estimator has a very good behavior for small-to-moderate sample sizes. It deteriorates in performance compared to fixed bandwidth estimates as the sample size grows.

2.3 Balloon estimator

The balloon estimator was first introduced by Loftsgaarden and Quensberry [14]. It is defined as:

$$\begin{aligned}\widehat{f}(\mathbf{x}) &= \frac{1}{n}\sum_{i=1}^{n} K_{\mathbf{H}(\mathbf{x})}(\mathbf{x}-\mathbf{x}^{(i)}) \\ &= \frac{1}{n}\sum_{i=1}^{n} \frac{1}{|\mathbf{H}(\mathbf{x})|^{1/2}} K(\mathbf{H}(\mathbf{x})^{-1/2}(\mathbf{x}-\mathbf{x}^{(i)})) \ .\end{aligned} \tag{5}$$

This estimator allows a straightforward asymptotic analysis since it uses standard pointwise results [15]. On the other hand, when applied globally, the estimate typically does not integrate to 1 and thus is usually not itself a density, even when K is. In [23] the authors have investigated the improvement that this estimator allows over fixed bandwidth kernel estimates. For data of fewer than 3 dimensions, the improvement seems to be very modest. However the balloon estimator becomes very efficient as soon as the number of dimensions becomes higher than 3.

2.4 Quality of an estimator

The quality of an estimator depends on the closeness of $\widehat{f}$ to the target density f. A common measure of this closeness is the mean squared error (MSE), equal to the sum of the variance and the squared bias:

$$\begin{aligned} \mathrm{MSE}(\mathbf{x}) &= E\big[(\widehat{f}(\mathbf{x}) - f(\mathbf{x}))^2\big] \\ &= \mathrm{var}(\widehat{f}(\mathbf{x})) + \big[\mathrm{Bias}(\widehat{f}(\mathbf{x}))\big]^2 \ . \end{aligned} \tag{6}$$

A good estimator has a small bias and a small variance. We detail in this subsection the computation of the bias and the variance for the fixed bandwidth estimator. For that purpose, we denote as ∇f the gradient of function f and as $\mathcal{H}(f)$ the Hessian matrix of second partial derivatives. The second-order Taylor expansion of $f(\bullet)$ around $\mathbf{x}$ [24, p.94] is:

$$f(\mathbf{x}+\delta\mathbf{x}) = f(\mathbf{x}) + \delta\mathbf{x}^T\nabla f(\mathbf{x}) + \frac{1}{2}\delta\mathbf{x}^T\mathcal{H}(f(\mathbf{x}))\delta\mathbf{x} + o(\delta\mathbf{x}^T\delta\mathbf{x}) \ . \tag{7}$$

Applying to the fixed kernel estimator, it leads to the expectation:

$$\begin{aligned} E\big(\widehat{f}(\mathbf{x})\big) &= \int \frac{1}{|\mathbf{H}|^{1/2}} K(\mathbf{H}^{-1/2}(\mathbf{u}-\mathbf{x}))f(\mathbf{u})\mathrm{d}\mathbf{u} \\ &= \big[\int K(\mathbf{s})\mathrm{d}\mathbf{s} f(\mathbf{x}) + \int K(\mathbf{s})\mathbf{s}^T\mathrm{d}\mathbf{s}\ {\mathbf{H}^{1/2}}^T\nabla f(\mathbf{x}) + \\ &\qquad \int \frac{1}{2}K(\mathbf{s})(\mathbf{H}^{1/2}\mathbf{s})^T\mathcal{H}(f(\mathbf{x}))(\mathbf{H}^{1/2}\mathbf{s})\mathrm{d}\mathbf{s} + \int K(\mathbf{s})o(\mathbf{s}^T\mathbf{H}\mathbf{s})\mathrm{d}\mathbf{s}\big] \ . \end{aligned} \tag{8}$$

Using the kernel properties (equation 6), the fact that the trace of a scalar is just the scalar, and the identity $\mathrm{tr}(\mathbf{AB}) = \mathrm{tr}(\mathbf{BA})$, the bias of the fixed kernel estimator becomes:

$$\begin{aligned} \mathrm{Bias}(\widehat{f}(\mathbf{x})) &= E\big(\widehat{f}(\mathbf{x})\big) - f(\mathbf{x}) \\ &= c_k \mathrm{tr}\big[{\mathbf{H}^{1/2}}^T\mathcal{H}(f(\mathbf{x}))\mathbf{H}^{1/2}\big] + \int K(\mathbf{s})o(\mathbf{s}^T\mathbf{H}\mathbf{s})\mathrm{d}\mathbf{s} \ . \end{aligned} \tag{9}$$

The variance is

$$\begin{aligned} \mathrm{var}(\widehat{f}(\mathbf{x})) &= \mathrm{var}\big[\frac{1}{n}\sum_{i=1}^{n} K_{\mathbf{H}}(\mathbf{x}-\mathbf{x}^{(i)})\big] \\ &= \frac{1}{n|\mathbf{H}|^{1/2}}\big(\int (K(\mathbf{s}))^2\mathrm{d}\mathbf{s} f(\mathbf{x}) + \int K(\mathbf{s})o(\mathbf{s}^T\mathbf{H}\mathbf{s})\mathrm{d}\mathbf{s}\big) \ . \end{aligned} \tag{10}$$

Several other measures exist, mainly the mean integrated squared error (MISE) or the asymptotic mean integrated squared error (AMISE). A detailed derivation of these measures can be found in [19] and [24]. As discussed in 4.1, these measures can be used to select the best value for $\mathbf{H}$.

3 Mean shift partitioning

An appealing technique for clustering is the mean shift algorithm, which does not require to fix the (maximum) number of clusters. In this section we first remind the definition of kernel profiles and

the principle of mean shift filtering and partitioning. As we are interested in variable bandwidth estimation, we give the result of [7] in which the mean shift for the sample point estimator was developed. We then introduce a novel pseudo balloon mean shift based on the balloon estimator.

3.1 Kernel profile

The profile of a kernel K is the function $k : [0, \infty) \to \mathbb{R}$ such that $K(x) = c_k k(\|x\|^2)$, where c_k is a positive normalization constant which makes $K(\mathbf{x})$ integrate to one. Using this profile the fixed bandwidth kernel density estimator can be rewritten as:

$$\begin{aligned} \widehat{f}(\mathbf{x}) &= \frac{1}{n|\mathbf{H}|^{1/2}} \sum_{i=1}^{n} K(\mathbf{H}^{-1/2}(\mathbf{x} - \mathbf{x}^{(i)})) \\ &= \frac{c_k}{n|\mathbf{H}|^{1/2}} \sum_{i=1}^{n} k(\|\mathbf{H}^{-1/2}(\mathbf{x} - \mathbf{x}^{(i)})\|^2) \end{aligned} \tag{11}$$

Two main kernels are used for mean shift filtering. Using a fixed bandwidth estimator, it can be shown [19, p.139][24, p.104] that the AMISE measure is minimized by the Epanechnikov kernel having the profile:

$$k_E(x) = \begin{cases} 1 - x & 0 \leq x \leq 1 \\ 0 & x > 1 \ . \end{cases} \tag{12}$$

The drawback of the Epanechnikov kernel is that it is not differentiable at the boundary of its support (for $x = 1$). The second kernel is the multivariate normal one defined by the profile:

$$k(x) = \exp(-\frac{1}{2}x) \tag{13}$$

which leads to the interesting property:

$$g(x) = -k'(x) = -\frac{1}{2}k(x) \ . \tag{14}$$

The normalization for this profile is $c_k = (2\pi)^{-d/2}$.

3.2 Fixed bandwidth mean shift filtering and partitioning

Mean shift is an iterative gradient ascent method used to locate the density modes of a cloud of points, *i.e.* the local maxima of its density. The mean shift filtering is well described in [5]. Here the theory is briefly reminded.

The density gradient of the fixed kernel estimator (equation 1) is given by:

$$\nabla \widehat{f}(\mathbf{x}) = \mathbf{H}^{-1} \, \widehat{f}(\mathbf{x}) \, \mathbf{m}(\mathbf{x}) \tag{15}$$

where $\mathbf{m}$ is the "mean shift" vector,

$$\mathbf{m}(\mathbf{x}) = \frac{\sum_{i=1}^{n} \mathbf{x}^{(i)} \, g\big(\|\mathbf{H}^{-1/2}(\mathbf{x} - \mathbf{x}^{(i)})\|^2\big)}{\sum_{i=1}^{n} g\big(\|\mathbf{H}^{-1/2}(\mathbf{x} - \mathbf{x}^{(i)})\|^2\big)} \ - \ \mathbf{x} \quad . \tag{16}$$

Using exactly this displacement vector at each step guaranties convergence to the local maximum of the density [5]. A mode seeking algorithm (algorithm 1), or mean shift filtering can be derived by iteratively computing the mean shift vector. Each computation of this vector leads to a trajectory point $\mathbf{y}^{(j)}$. The first trajectory point $\mathbf{y}^{(1)}$ is the estimation point $\mathbf{x}$ itself while the last point $\mathbf{y}^{(t_m)}$ is the associated mode $\mathbf{z}$. The final partition of the feature space is obtained by grouping together all the data points that converged to the same mode (algorithm 2).

Algorithm 1 Mean shift filtering

Let $\{\mathbf{x}^{(i)}\}_{i=1,\ldots,n}$ be n input points in the d-dimensional space and $\{\mathbf{z}^{(i)}\}_{i=1,\ldots,n}$ their associated modes. For $i = 1 \ldots n$

1. Initialize $j = 1$, $\mathbf{y}^{(1)} = \mathbf{x}^{(i)}$.
2. Repeat
   - $\mathbf{y}^{(j+1)} = \mathbf{y}^{(j)} + \mathbf{m}(\mathbf{y}^{(j)})$ according to equation 16.
   - $j = j + 1$.

   Until $\mathbf{y}^{(j-1)} = \mathbf{y}^{(j)}$.
3. Assign $\mathbf{z}^{(i)} = \mathbf{y}^{(j)}$.

Algorithm 2 Mean shift partitioning

Let $\{\mathbf{x}^{(i)}\}_{i=1,\ldots,n}$ be n input points in the d-dimensional space and $\{\mathbf{z}^{(i)}\}_{i=1,\ldots,n}$ their associated modes.

1. Run the mean shift filtering algorithm.
2. Group together all $\mathbf{z}^{(i)}$ which are closer than $\mathbf{H}$, *i.e* two modes $\mathbf{z}^{(i)}$ and $\mathbf{z}^{(j)}$ are grouped together if :

$$\|\mathbf{z}^{(i)} - \mathbf{z}^{(j)}\| \leq \|\mathbf{H}\| \ .$$

3. Group together all $\mathbf{x}^{(i)}$ whose associated mode belongs to the same group.

Mean shift with normal kernel usually needs more iterations to converge, but yields results that are almost always better than the ones obtained with the Epanechnikov kernel. In the sequel we will only consider the multivariate normal kernel. With a *d*-variate Gaussian kernel, equation 16 becomes

$$\mathbf{m}(\mathbf{x}) = \frac{\sum_{i=1}^{n} \mathbf{x}^{(i)} \ \exp(-\frac{1}{2} D^2(\mathbf{x}, \mathbf{x}^{(i)}, \mathbf{H}))}{\sum_{i=1}^{n} \exp(-\frac{1}{2} D^2(\mathbf{x}, \mathbf{x}^{(i)}, \mathbf{H}))} \ - \ \mathbf{x} \tag{17}$$

where

$$D^2(\mathbf{x}, \mathbf{x}^{(i)}, \mathbf{H}) \equiv (\mathbf{x} - \mathbf{x}^{(i)})^T \mathbf{H}^{-1} (\mathbf{x} - \mathbf{x}^{(i)}) \tag{18}$$

is the squared Mahalanobis distance from $\mathbf{x}$ to $\mathbf{x}^{(i)}$.

3.3 Variable bandwidth mean shift

In the sequel we detail the mean shift using the two variable bandwidth estimators. The first version, called "variable bandwidth mean shift", is based on the sample point estimator and was introduced in [7]. The second one is novel, since the balloon estimator has never been used in a mean shift algorithm. We will refer to the algorithm as "pseudo balloon mean shift".

3.3.1 Variable bandwidth mean shift using sample point estimator

The mean shift filtering using the sample point estimator was first introduced in [7]. Using this estimator, equation 17 becomes:

$$\mathbf{m}(\mathbf{x}) = \frac{\sum_{i=1}^{n} |\mathbf{H}(\mathbf{x}^{(i)})|^{-1/2} \mathbf{x}^{(i)} \ \exp(-\frac{1}{2} D^2(\mathbf{x}, \mathbf{x}^{(i)}, \mathbf{H}(\mathbf{x}^{(i)})))}{\sum_{i=1}^{n} |\mathbf{H}(\mathbf{x}^{(i)})|^{-1/2} \exp(-\frac{1}{2} D^2(\mathbf{x}, \mathbf{x}^{(i)}, \mathbf{H}(\mathbf{x}^{(i)})))} \ - \ \mathbf{x} \ . \tag{19}$$

As with fixed bandwidth kernel estimator, a mean shift filtering algorithm can be derived based on this mean shift vector. The proof of convergence of mean shift filtering using the sample point estimator can be found in [7].

3.3.2 Pseudo balloon mean shift

As mentioned earlier, the balloon estimator $\widehat{f}(\mathbf{x})$ is not always a density (does not always integrate to one), and leads to discontinuity problems. Its derivative $\nabla \widehat{f}(\mathbf{x})$ contains terms that depend on $(\mathbf{x} - \mathbf{x}^{(i)})^2$ and of $\mathbf{H}'(\mathbf{x})$. Thus there is no closed-form expression for the mean shift vector. To be able to develop a mean shift filtering algorithm based on the balloon estimator, several assumptions must be made. In the context of mean shift algorithms, the bandwidth function $\mathbf{H}$ is only defined discretely at estimation points. To turn the estimator into a density and to give a closed form to the derivatives, we assume that $\forall i = 1 \ldots n, \mathbf{H}'(\mathbf{x}^{(i)}) = 0$. Using the kernel profile k, equation 5 evaluated at data points becomes, for $i = 1 \ldots n$:

$$\widehat{f}(\mathbf{x}^{(i)}) = \frac{c_k}{n} \sum_{j=1}^{n} \frac{1}{|\mathbf{H}(\mathbf{x}^{(i)})|^{1/2}} k(\|\mathbf{H}(\mathbf{x}^{(i)})^{-1/2} (\mathbf{x}^{(j)} - \mathbf{x}^{(i)})\|^2) \ . \tag{20}$$

Since we consider $\mathbf{H}'(\mathbf{x}^{(i)}) = 0$, its derivative is:

$$\begin{aligned}
\widehat{\nabla} f(\mathbf{x}^{(i)}) &= \nabla \widehat{f}(\mathbf{x}^{(i)}) \\
&= \frac{c_k}{n|\mathbf{H}(\mathbf{x}^{(i)})|^{1/2}} \sum_{j=1}^{n} \mathbf{H}(\mathbf{x}^{(i)})^{-1}(\mathbf{x}^{(j)} - \mathbf{x}^{(i)}) k(\|\mathbf{H}(\mathbf{x}^{(i)})^{-1/2}(\mathbf{x}^{(j)} - \mathbf{x}^{(i)})\|^2) \\
&= \frac{c_k}{n|\mathbf{H}(\mathbf{x}^{(i)})|^{1/2}} \mathbf{H}(\mathbf{x}^{(i)})^{-1} \sum_{j=1}^{n} k(\|\mathbf{H}(\mathbf{x}^{(i)})^{-1/2}(\mathbf{x}^{(j)} - \mathbf{x}^{(i)})\|^2)(\mathbf{x}^{(j)} - \mathbf{x}^{(i)}) \\
&= \frac{1}{n}\Big[\sum_{i=1}^{n} \mathbf{H}(\mathbf{x}^{(i)})^{-1} K_{\mathbf{H}(\mathbf{x}^{(i)})}(\mathbf{x}^{(j)} - \mathbf{x}^{(i)})\Big]\Big[\frac{\sum_{j=1}^{n} \mathbf{x}^{(j)} K_{\mathbf{H}(\mathbf{x}^{(i)})}(\mathbf{x}^{(j)} - \mathbf{x}^{(i)})}{\sum_{j=1}^{n} K_{\mathbf{H}(\mathbf{x}^{(i)})}(\mathbf{x}^{(j)} - \mathbf{x}^{(i)})} - \mathbf{x}^{(i)}\Big] \ .
\end{aligned} \tag{21}$$

A mean shift filtering algorithm can be derived using the last term of previous equation as the mean shift vector:

$$\mathbf{m}(\mathbf{x}) = \frac{\sum_{j=1}^{n} \mathbf{x}^{(j)} K_{\mathbf{H}(\mathbf{x})}(\mathbf{x} - \mathbf{x}^{(j)})}{\sum_{j=1}^{n} K_{\mathbf{H}(\mathbf{x})}(\mathbf{x} - \mathbf{x}^{(j)})} - \mathbf{x} \ . \tag{22}$$

Previous equation is only valid at the data points. Therefore, if $\mathbf{H}$ varies for each trajectory point, the mean shift filtering algorithm is not valid and its convergence can not be proved. The solution that we propose is to defined a pseudo balloon mean shift where the bandwidth varies for each estimation point but is fixed for all trajectory points. This means that the data points influencing the computation of a trajectory point are taken with the same bandwidth along all the gradient ascent trajectory. The advantage is that the estimate at a point will not be influenced by observations too far away. We then take the bandwidth $\mathbf{H}(\mathbf{x})$ constant for all trajectory points $\mathbf{y}^{(j)}$ corresponding to the estimation point $\mathbf{x}$ (belonging to the data points). In other words, for a given starting point, the procedure amounts to a fixed bandwidth mean shift, with bandwidth depending on the starting point. We call this new mean shift pseudo balloon mean shift. The convergence of the pseudo balloon mean shift filtering if $\mathbf{H}(\mathbf{x})^T = \mathbf{H}(\mathbf{x})$ is demonstrated in appendix. The pseudo balloon mean shift partitioning algorithm is described in Algorithm 3. We use the minimum of the two bandwidths in step 2 to avoid the aggregation of two very distant modes.

4 Bandwidth selection for mixed feature spaces

Results of mean shift filtering or partitioning always highly depend on the kernel bandwidth $\mathbf{H}$ which has to be chosen carefully. Various methods for bandwidth selection exist in literature. In particular, several statistical criteria, which generally aim at balancing the bias against the variance of an estimator, have been introduced. They are called statistical-analysis based methods, and can be applied to any method based on kernel estimation. Other techniques, only dedicated to clustering, define a criteria based on the stability of the clusters. They are called stability based methods. In subsection 4.1, we present a review of these two types of techniques. Many of these methods have proven to be

Algorithm 3 Pseudo balloon mean shift partitioning algorithm

Let $\{\mathbf{x}^{(i)}\}_{i=1,\dots,n}$ be n input points in the d-dimensional space and $\{\mathbf{z}^{(i)}\}_{i=1,\dots,n}$ their associated modes.

1. For $i = 1, \dots, n$, run the mean shift filtering algorithm from $\mathbf{x} = \mathbf{x}^{(i)}$, with $\mathbf{H}(\mathbf{x}) = \mathbf{H}(\mathbf{x}^{(i)})$, using equation (22).
2. Group together two modes $\mathbf{z}^{(i)}$ and $\mathbf{z}^{(j)}$ if:
$$\|\mathbf{z}^{(i)} - \mathbf{z}^{(j)}\| \leq \min(\|\mathbf{H}(\mathbf{x}^{(i)})\|, \|\mathbf{H}(\mathbf{x}^{(j)})\|) \ .$$
3. Group together all $\mathbf{x}^{(i)}$ whose associated modes belong to the same group.

very efficient. Nevertheless, none of them is really adapted to data in high dimensional heterogeneous spaces.

Therefore, in this section we propose an algorithm dedicated to the mean shift partitioning in high dimensional heterogeneous space. We assume that the d-dimensional input space can be decomposed as the Cartesian product of P independent spaces associated to different types of information (*e.g.* position, color), also called feature spaces or domains, with dimension $d_\rho, \rho = 1 \dots P$ (where $\sum_{\rho=1}^{P} d_\rho = d$).

4.1 Existing methods for bandwidth selection

This first subsection present a short review of existing bandwidth selection methods of both types.

4.1.1 Statistical-analysis based methods

Statistical methods aim at improving the quality of the kernel estimator. We remind that a good estimator is an estimator that has a small bias and a small variance. The quality is usually evaluated by measuring the distance (MSE, MISE, AMISE...) between the estimate $\widehat{f}$ and the target density f. These measures are of little practical use since they depend on the unknown density function f both in the variance and the squared bias term (subsection 2.4). However, the definition of the bias and the variance leads to the following property: the absolute value of the bias increases and the variance decreases as $\mathbf{H}$ increases. Therefore to minimize the mean squared error (or any other measure), we are confronted with a bias-variance trade-off.

Several good solutions can be found in literature to find the best value for $\mathbf{H}$. In particular, we can mention the "rule of thumb" method [21], the "plug-in" rules [17, 20] and the cross validation methods [17][22, p.46]. However, all these techniques have some drawbacks. The "rule of thumb" assumes that the density is Gaussian. A practical algorithm based on the plug-in rule for one dimensional data can be found in [5]. An algorithm for the case of multivariate data is presented in [24, p.108] but it

is difficult to implement. Finally, cross validation methods becomes very computationally expensive for a large set of data.

For variable bandwidth, the most often used method takes the bandwidth proportional to the inverse of the square root of a first-order approximation of the local density. This is the Abramson's rule [1]. Two parameters must then be tuned in advance: a proportionality constant and an initial fixed bandwidth. The proportionality constant influences a lot the result. Also, for multidimensional multimodal data, the initial fixed bandwidth is hard to determine. The application of this technique to the variable bandwidth mean shift, *i.e.* to the sample point estimator, has been proposed and discussed in [7]. It leads to good results on toy examples. Evaluation of partitioning on real data is subjective, and it is hard to assert the superiority of such statistical methods. Furthermore, their application to high dimensional multimodal data is still an open problem.

4.1.2 A stability based method

Methods for bandwidth selection specially dedicated to clustering have also been studied. They are based on cluster validation. Many criteria determining the validity of a cluster exist [16]. For example, some methods evaluate the isolation and connectivity of the clusters. Another criterion is the stability. It is based on human visual experience: real clusters should be perceivable over a wide range of scales. This criterion has been used in the scale space theory in [13] or in [8] where the stability of clusters depends on the size of the clusters. An application of a stability criterion to bandwidth selection for mean shift partitioning was introduced in [4]. The idea is that a partition should not change when a small variation is applied to the bandwidth.

As no statistical methods are currently well adapted to the variable bandwidth estimation in high dimensional heterogeneous data, we decided to use the stability to validate the partitions and to find the best bandwidth at each point. The basic principle of the method that we propose is based on the one in [4]. The goal is to find the best bandwidth at each point within a set of predefined bandwidths. Given a set of B predefined matrices $\{\mathbf{H}^{(b)}, b = 1, \ldots, B\}$, the best bandwidth, denoted as $\Upsilon(\mathbf{x}^{(i)})$, in this predefined set, at each point $\mathbf{x}^{(i)}$ indexed by i, is the one that gives the most stable clusters. The method is composed of two main steps.

The first step is called bandwidth evaluation at the partition level. The mean shift partitioning is run for each of the predefined matrices. For each scale b of this range, the data is divided into a certain number of clusters. For simplicity we introduce the function c which, for each scale b, associates a data point indexed by i to its corresponding cluster. If the i-th data point belongs to the u-th cluster at scale b, then $c(i, b) = u$. Each cluster is then represented parametrically. Indeed the stability criteria chosen in [4] asserts that if the clusters can be represented by normal laws, then the best cluster is the one for which the normal law is the most stable. If few points are added to the partition or if some are left apart, the distribution of the cluster should not change. The assumption of normality seems

reasonable in a small neighborhood of a point, this neighborhood being found by the partitioning. Each cluster indexed by u at scale b is then represented parametrically by a normal law $\mathcal{N}(\mu_u^{(b)}, \mathbf{\Sigma}_u^{(b)})$. Let $\mathcal{C}_u^{(b)}$ be the set of indices of points belonging to a cluster u at scale b, $\mathcal{C}_u^{(b)} = \{i/c(i,b) = u\}$. The mean $\mu_u^{(b)}$ corresponding to cluster u at scale b is defined as:

$$\mu_u^{(b)} = \frac{1}{|\mathcal{C}_u^{(b)}|} \sum_{i \in \mathcal{C}_u^{(b)}} \mathbf{x}^{(i)} \ , \tag{23}$$

and the empirical covariance $\mathbf{\Sigma}_u^{(b)}$ as:

$$\mathbf{\Sigma}_u^{(b)} = \frac{1}{|\mathcal{C}_u^{(b)}|} \sum_{i \in \mathcal{C}_u^{(b)}} (\mathbf{x}^{(i)} - \mu_u^{(b)})(\mathbf{x}^{(i)} - \mu_u^{(b)})^T \ . \tag{24}$$

These expectation and covariance estimates are easily corrupted by non-Gaussian tails which might occur. In [4], other formula have been established to solve this problem. However, the proposed covariance does not seem reliable since it can be negative. Therefore, in the sequel, we will consider that all the means and covariances are computed with the traditional definitions. This choice gave satisfactory results for all the tests we have run but we believe that further work should concentrate on finding a better way to compute the covariance based on existing techniques for robust covariance matrix estimation [18, 25].

After building all the normal laws, each point is associated at each scale to the law of the cluster it belongs to. The point indexed by i is associated for scale b to the distribution $p_i^{(b)} = \mathcal{N}(\mu_{c(i,b)}^{(b)}, \mathbf{\Sigma}_{c(i,b)}^{(b)})$.

The second step evaluates for each point the clusters to which this point was associated and finds the most stable one. This second step is called bandwidth evaluation at the data level. It mainly consists in the comparison of the clusters, through the comparison of the normal laws. Several divergence measures between multiple probability distributions have been studied in literature. In [4], the authors use the Jensen-Shannon divergence to compare the distributions. Given r d-variate normal distributions $p_j \ , \ j = 1, \ldots, r$, defined by their mean μ_j and covariance $\mathbf{\Sigma}_j$, the Jensen-Shannon divergence is defined as:

$$JS(p_1 \ldots p_r) = \frac{1}{2} \log \frac{|\frac{1}{r}\sum_{j=1}^r \mathbf{\Sigma}_j|}{\sqrt[r]{\prod_{j=1}^r |\mathbf{\Sigma}_j|}} + \frac{1}{2} \sum_{j=1}^r (\mu_j - \frac{1}{r}\sum_{j=1}^r \mu_j)^T (\sum_{j=1}^r \mathbf{\Sigma}_j)^{-1} (\mu_j - \frac{1}{r}\sum_{j=1}^r \mu_j) \ . \tag{25}$$

The comparison is done between three neighboring scales ($r = 3$) and for each domain independently, the distributions being p_i^{b-1}, p_i^b and p_i^{b+1}. The best scale $b^* = \text{argmin}_b JS(p_i^{(b-1)}, p_i^{(b)}, p_i^{(b+1)})$ is the one for which the Jensen-Shannon divergence,

$$\begin{aligned} JS(p_i^{(b-1)}, p_i^{(b)}, p_i^{(b+1)}) = & \frac{1}{2} \log \frac{|\frac{1}{3}\sum_{j=b-1}^{b+1} \mathbf{\Sigma}_{c(i,b)}^{(j)}|}{\sqrt[3]{\prod_{j=b-1}^{b+1} |\mathbf{\Sigma}_{c(i,b)}^{(j)}|}} + \\ & \frac{1}{2} \sum_{j=b-1}^{b+1} (\mu_{c(i,b)}^{(j)} - \frac{1}{3}\sum_{j=b-1}^{b+1} \mu_{c(i,b)}^{(j)})^T (\sum_{j=b-1}^{b+1} \mathbf{\Sigma}_{c(i,b)}^{(j)})^{-1} (\mu_{c(i,b)}^{(j)} - \frac{1}{3}\sum_{j=b-1}^{b+1} \mu_{c(i,b)}^{(j)}) \ , \end{aligned} \tag{26}$$

is minimized. The final best bandwidth for the point $\mathbf{x}^{(i)}$ is the predefined matrix that gave the most stable cluster: $\boldsymbol{\Upsilon}(\mathbf{x}^{(i)}) = \mathbf{H}^{(b^*)}$. This is in the contrast with the original method in [4], where $\boldsymbol{\Upsilon}(\mathbf{x}^{(i)}) = \boldsymbol{\Sigma}^{(b^*)}_{c(i,b^*)}$. The latter choice does not guarantee that the estimated bandwidth lies between the extremal bandwidths of the selected range ($\mathbf{H}^{(1)}$ and $\mathbf{H}^{(B)}$ when they are sorted).

These two steps are described in Figure 1. The final partition of the data is obtained by rerunning a variable or a pseudo balloon mean shift partitioning using the selected matrices. Unfortunately, this algorithm is limited to data composed of one feature space. Therefore, in next subsection we propose an iterative algorithm, based on the previous method, that handles the heterogeneity of the data.

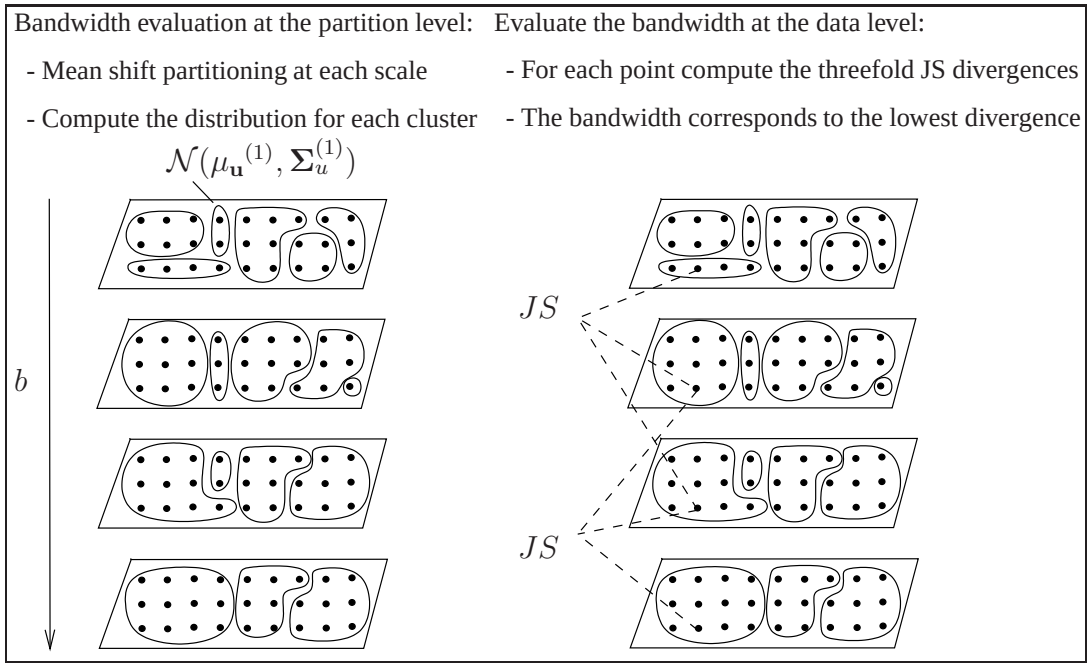


Figure 1: Scheme of an iteration of our algorithm.

4.2 Handling heterogeneity: iterative selection

For high dimensional heterogeneous data the set of predefined bandwidths can become large. Indeed, if the different domains of multidimensional data are independent, the bandwidth for each feature space should be chosen independently. The most stable cluster is not obtained for the same scale for all the domains. A solution could be to define a set of bandwidths for each feature space and to partition the data using all the possible bandwidth matrices resulting from the combination of the different sets. However as the dimensions become high and/or the sets of predefined bandwidths

become larges, the algorithm can become computationally very expensive: if we have a range of B_ρ analysis bandwidths for each feature space ρ, the mean shift partitioning has to be run $\prod_{\rho=1}^{P} B_\rho$ times to take into account every possibility. Thus the algorithm is not adapted for spaces composed of several independent components. The solution is then to find the best bandwidth iteratively for each feature space, so that the mean shift partitioning is run only $\sum_{\rho=1}^{P} B_\rho$ times.

Suppose that we are trying to select the best bandwidth at each data point for the first feature space. We fix temporary matrices $\tilde{\mathbf{H}}_\rho, \rho = 2, \ldots, P$ for each of the other feature spaces. These matrices are constant for all scales and equal to the mean over all the B_ρ possible matrices:

$$\tilde{\mathbf{H}}_\rho = \frac{1}{B_\rho} \sum_{b=1}^{B_\rho} \mathbf{H}_\rho^{(b)}, \rho > 1 \ . \tag{27}$$

The bandwidth selection algorithm previously defined (Figure 1) is run for the matrix range

$$\{\tilde{\mathbf{H}}^{(b)} = \mathrm{diag}[\mathbf{H}_1^{(b)}, \tilde{\mathbf{H}}_2, \ldots, \tilde{\mathbf{H}}_P], b = 1, \ldots, B_1\}$$

and finds the best bandwidth $\boldsymbol{\Upsilon}_1(\mathbf{x}^{(i)})$ for each point $\mathbf{x}^{(i)}$. The same procedure is then run for every other feature space. The difference is that for the feature spaces that have already been studied the bandwidth matrix is not constant anymore:

$$\tilde{\mathbf{H}}^{(b)}(\mathbf{x}^{(i)}) = \mathrm{diag}[\boldsymbol{\Upsilon}_1(\mathbf{x}^{(i)}), \ldots, \boldsymbol{\Upsilon}_{\rho-1}(\mathbf{x}^{(i)}), \mathbf{H}_\rho^{(b)}, \tilde{\mathbf{H}}_{\rho+1} \ldots \tilde{\mathbf{H}}_P] \ . \tag{28}$$

A variable bandwidth mean shift must then be used. As the dimension of the data is higher than 3, we prefer the balloon based mean shift partitioning, but the sample point estimator could be used as well within the same procedure.

4.3 Bandwidth selection final algorithm

The proposed iterative algorithm solves the bandwidth selection for high dimensional heterogeneous data problem. Each feature space is processed successively in two stages. The first stage consists in partitioning the data for each scale and building a parametric representation of each cluster. The second stage selects for each data point the most stable cluster which finally leads to the best bandwidth. The final algorithm is presented in algorithm 4.

5 Experimental results

This section presents some results of our method on color image segmentation. The final partition of the data is obtained by applying a last time the pseudo balloon mean shift partitioning with the selected variable bandwidths. The data set is the set of all pixels of the image. To each data point is associated a 5-dimensional feature vector: two dimensions for the position and three for the color.

Algorithm 4 Iterative estimation of bandwidths

Given a set of B_ρ predefined bandwidths $\{\mathbf{H}_\rho^{(b)}, b = 1 \dots B\}$ for each feature space ρ. The bandwidth selection is as follows.

For $\rho = 1, \dots, P$

- Evaluate the bandwidth at the partition level: For all $b = 1, \dots, B$
  1. For all $\rho' = \rho + 1, \dots, P$, compute $\tilde{\mathbf{H}}_{\rho'}$:
  $$\tilde{\mathbf{H}}_{\rho'} = \frac{1}{B_{\rho'}} \sum_{b=1}^{B_{\rho'}} \mathbf{H}_{\rho'}^{(b)} \quad . \tag{29}$$
  2. Define, for $i = 1, \dots, n$,
  $$\{\tilde{\mathbf{H}}^{(b)}(\mathbf{x}^{(i)}) = \mathrm{diag}[\boldsymbol{\Upsilon}_1(\mathbf{x}^{(i)}), \dots, \boldsymbol{\Upsilon}_{\rho-1}(\mathbf{x}^{(i)}), \mathbf{H}_\rho^{(b)}, \tilde{\mathbf{H}}_{\rho+1} \dots \tilde{\mathbf{H}}_P], b = 1, \dots, B_\rho\}.$$
  3. Partition the data using the balloon mean shift partitioning. The result is $n^{(b)}$ clusters denoted as $\mathcal{C}_u^{(b)}, u = 1 \dots n^{(b)}$. Introduce the function c that associates a point indexed by i to its cluster u: $c(i, b) = u$.
  4. Compute the parametric representation $\mathcal{N}(\mu_u^{(b)}, \boldsymbol{\Sigma}_u^{(b)})$ of each partition using:
  $$\mu_u^{(b)} = \frac{1}{|\mathcal{C}_u^{(b)}|} \sum_{i \in \mathcal{C}_u^{(b)}} \mathbf{x}^{(i)} = \begin{bmatrix} \mu_{u,1}^{(b)} \\ \vdots \\ \mu_{u,P}^{(b)} \end{bmatrix} \quad , \tag{30}$$
  and
  $$\boldsymbol{\Sigma}_u^{(b)} = \frac{1}{|\mathcal{C}_u^{(b)}|} \sum_{i \in \mathcal{C}_u^{(b)}} (\mathbf{x}^{(i)} - \mu_u^{(b)})(\mathbf{x}^{(i)} - \mu_u^{(b)})^T = \mathrm{diag}[\boldsymbol{\Sigma}_{u,1}^{(b)}, \dots, \boldsymbol{\Sigma}_{u,P}^{(b)}] \quad . \tag{31}$$
  5. Associate to each point $\mathbf{x}^{(i)}$ the mean $\mu_{c(i,b),\rho}^{(b)}$ and covariance $\boldsymbol{\Sigma}_{c(i,b),\rho}^{(b)}$ of the cluster it belongs to and the corresponding normal distribution $p_{i,\rho}^{(b)}$.
- Evaluate the bandwidth at the data level: For each point $\mathbf{x}^{(i)}$
  1. Select the scale b^* giving the most stable normal distribution by solving:
  $$b^* = \mathrm{argmin}_{r=2,\dots,B-1} \mathrm{JS}(p_{i,\rho}^{(r-1)}, p_{i,\rho}^{(r)}, p_{i,\rho}^{(r+1)}) \tag{32}$$
  where JS is the Jensen-Shanon divergence defined by equation (26).
  2. The best bandwidth $\Upsilon_\rho(\mathbf{x}^{(i)})$ is $\mathbf{H}_\rho^{(b^*)}$.

We here consider the independency of all the dimensions, *i.e.* 5 features spaces each composed of one dimension. The order in which the dimensions are processed by our algorithm is the following: x coordinate, y coordinate, red, green and blue channels. In the final subsection we discuss the influence of the order in which the feature spaces are processed. For all the results presented in this section the

same predefined bandwidths are used. For all the feature spaces, we used 9 predefined bandwidths in the range of 10-30. Of course this range is large and it would be better to adapt it to each image, for example by using some information on the noise as in [2], but it is sufficient to validate our algorithm. The color of a pixel in the segmented images corresponds to the color of its associated mode.

The two novel features of our algorithm are successively validated with comparisons to other methods. First we validate the iterative bandwidth selection in independent feature spaces by comparing our algorithm with the same method in which the bandwidths evolve jointly in the different feature spaces. We then compare the variable mean shift based on the sample point estimator and the pseudo balloon mean shift. Several results for each of these two points are shown.

5.1 Validation of the iterative bandwidth selection

We start by validating the iterative selection on several examples. The comparison is done between our algorithm with five feature spaces and our algorithm with a single five-dimensional feature space. In the last case, all dimensions are consider dependent and the same scale is finally selected for all dimensions.

The first results are presented on an outdoor image. Figure 2 shows the final partitioning for the non iterative selection Figure 2(b)) and the iterative selection Figure 2 (c)). With the non-iterative algorithm, 21 clusters were found, while the iterative method gave 31 clusters. At the end of the segmentation the sky and the mountains are merged together with the non iterative algorithm. Differences are also visible on the mountains, in which the iterative method gave more clusters. In figure

a) b) c)

Figure 2: Validation of the iterative selection on the outdoor image. a) Original image; b) Non iterative bandwidth selection; c) Iterative bandwidth selection.

3 the evolution through scales of the mean shift partitioning that corresponds to the first step of the non iterative algorithm (“evaluate the bandwidth at the partition level”) is shown. The evolution for our iterative algorithm is presented in figure 4. This time the evolution is shown through scales and through feature spaces. Because bandwidths evolve jointly in the different feature spaces with the non-iterative algorithm, many details are rapidly lost. Our algorithm allows more stability between two consecutive scales.

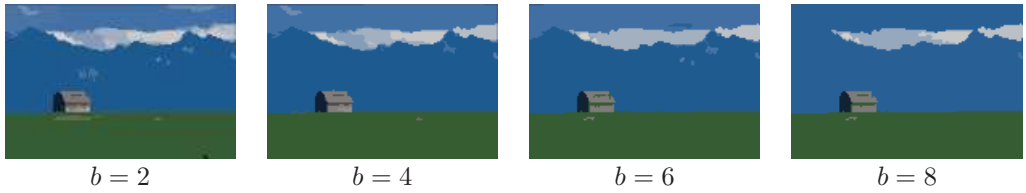


Figure 3: Evolution through scales of the partitionings for our non iterative algorithm.

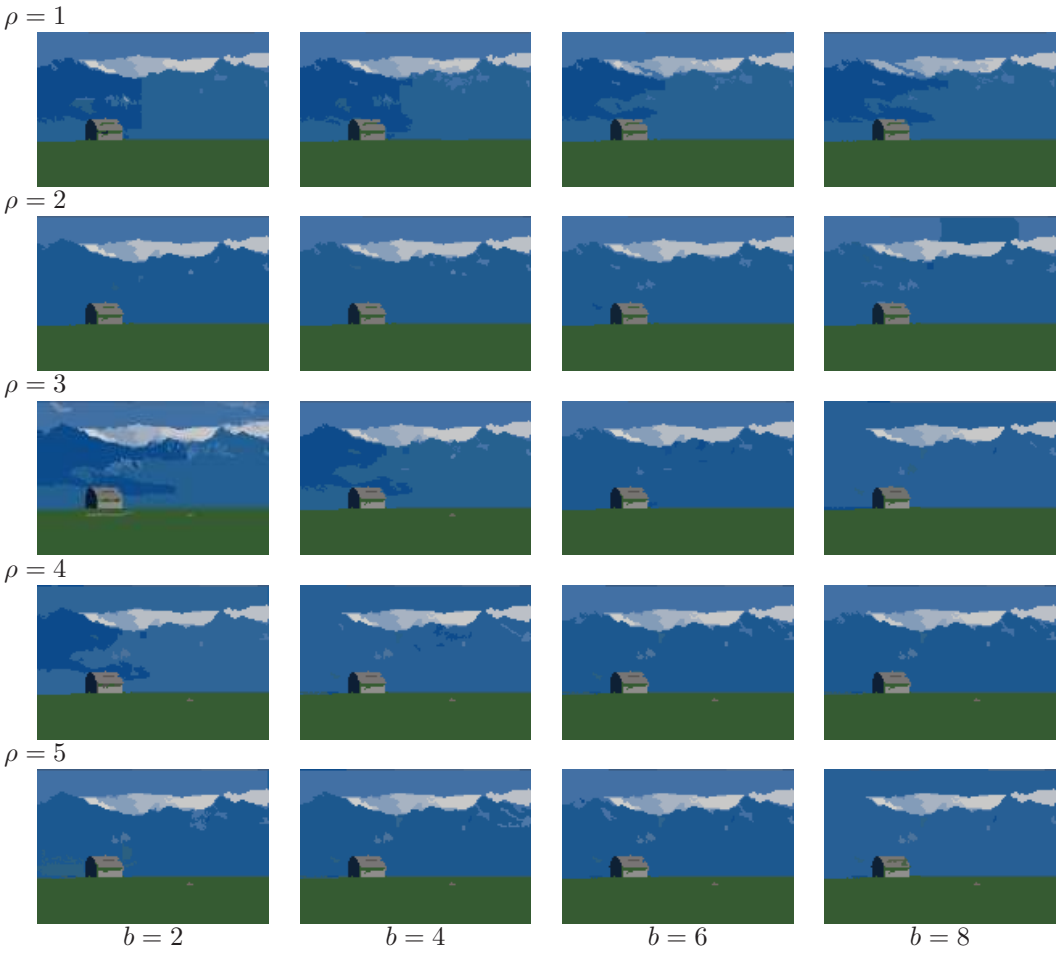


Figure 4: Evolution through scales and feature spaces of partitionings with our algorithm.

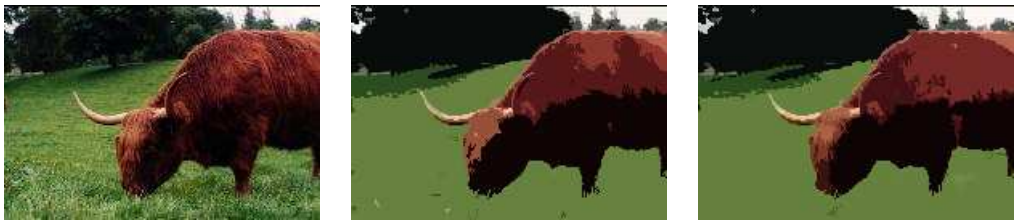
We show other segmentation results on the hand image (Figure 5 a)). For the first one 13 clusters are found against 37 for the second. The ring and the nails are not detected by the non iterative method because the selected color bandwidths are too large. The reason is that the regions to be

segmented are large, leading to large position bandwidths. Because the bandwidths for position and color are chosen jointly, the color bandwidth are also too large as a result.

a) b) c)

Figure 5: Validation of the iterative selection on the hand image. a) Original image; b) Non iterative bandwidth selection; c) Iterative bandwidth selection.

A last result is presented on the bull image (Figure 6 a)). Major differences between the segmentations obtained with the two methods are visible on the bull itself. In particular, the iterative algorithm keeps more details on the head of the animal. The number of clusters found by the two algorithms are not so different though. Indeed, the iterative method found 136 clusters while the non-iterative one gave 130 clusters. While more important details are kept on the bull by the iterative method, some little (but less important) clusters are lost on the grass.



a) b) c)

Figure 6: Validation of the iterative selection on the bull image. a) Original image; b) Non iterative bandwidth selection; c) Iterative bandwidth selection.

A surprising result concerns the computational cost. One could think that the non iterative method would be much faster than our iterative algorithm. This is not the case, even sometimes the iterative selection is faster. This can be explained as follows. While the first iterations are run for large bandwidths (mean over all the predefined bandwidths), in subsequent iterations, the best bandwidths have been chosen for the first feature spaces. These bandwidths are more adapted and lead to faster computation of the mean shift partitionings.

To conclude, the iterative method permits to keep more details than the non iterative one, even if the results are, sometimes, visually close. With the iterative method, it is not the same scale that is

chosen for all the dimensions. Furthermore, the introduction of our iterative selection method does not cause any computation overhead.

5.2 Validation of the pseudo balloon mean shift partitioning

A novelty of our approach is the introduction of the pseudo balloon mean shift partitioning. We compare this partitioning method to the variable mean shift partitioning introduced by Comaniciu in [7]. In [23] it has been shown that the balloon estimator gives good results when the number of dimensions increases, which led to its use in this paper. The comparison is done as follows. The bandwidth selection using the pseudo balloon mean shift partitioning as described in this paper is first run. Then using the selected bandwidths, the variable bandwidth and the pseudo balloon mean shift partitioning are run and give the final segmentations.

Here again we start by showing the results on the outdoor image. Figure 7 shows the final partitioning for the variable mean shift based on the sample point estimator (b)) and the pseudo balloon mean shift (c)). Nearly the same results were obtained by the two partitioning method. The sample point estimator gave 29 clusters against 31 for the pseudo balloon mean shift partitioning. Few tiny differences can be found in the clouds.

a) b) c)

Figure 7: Validation of "pseudo balloon mean shift" on the outdoor image. a) Original image; b) Variable mean shift partitioning; c) Pseudo balloon mean shift partitioning.

The next result is on the hand image (Figure 8). The segmentations are again very close with the two estimators. The variable mean shift partitioning [7] gave 35 clusters and the pseudo balloon 37. The segmentation of the forefinger for the variable sample point mean shift is slightly less clean (composed of two clusters).

We end this subsection by presenting the segmentation results on the bull image (Figure 9). Contrary to the two previous results, many differences are visible between the final partitioning obtained with the variable mean shift based on the sample point estimator (Figure 9(b)), which gave 128 clusters, and the pseudo balloon mean shift (c), which led to 136 clusters. The pseudo balloon mean shift permits to keep more details on the head of the bull. Also, the clusters found on the back are less messy than the ones found with the algorithm using the sample point estimator.

a) b) c)

Figure 8: Validation of the "pseudo balloon mean shift" on the hand image. a) Original image; b) Variable mean shift partitioning; c) Pseudo balloon mean shift partitioning.

a) b) c)

Figure 9: Validation of the "pseudo balloon mean shift" on the bull image. a) Original image; b) Variable mean shift partitioning; c) Pseudo balloon mean shift partitioning.

To conclude, the results presented in this subsection show that the pseudo balloon mean shift partitioning can be as good (or even better) as the variable mean shift of Comaniciu [7]. In addition, the balloon estimator is more adapted to high-dimensional ($d \geq 3$) heterogeneous data (*e.g.* five-dimensional data in color segmentation), thanks to the iterative bandwidth selection we introduced in 4.

5.3 Ordering the feature spaces

One could wonder if the order in which the feature spaces are studied is important. In fact it has only a small influence (Figures 10 and 11). These results have been obtained using 9 bandwidths in the range 3-20. As judging a segmentation depends on the subsequent application, defining the order in which the feature spaces should be processed is at this stage not really possible. An intuition would be to start with the position before processing successively the feature spaces having the highest noise or the highest contrast in the image.

6 Conclusion

Automatic bandwidth selection for kernel bandwidth estimation has become an important research area as the popularity of mean shift methods for image and video segmentation increases. Several

a) b)

Figure 10: Ordering the feature spaces. Results of the balloon mean shift partitioning on the outdoor image when the feature spaces are ordered in the following ways: a) x-coordinate, y-coordinate, red channel, green channel, blue channel; b) blue channel, green channel, red channel, y-coordinate, x-coordinate.

a) b)

Figure 11: Ordering the feature spaces. Results of the balloon mean shift partitioning on the baboon image when the feature spaces are ordered in the following ways: a) x-coordinate, y-coordinate, red channel, green channel, blue channel; b) blue channel, green channel, red channel, y-coordinate, x-coordinate.

methods already exist in literature but none of them is really adapted to the case of multidimensional heterogeneous features. This is the problem we addressed in this paper. To this end, we first introduced the pseudo balloon mean shift filtering and partitioning to which the kernel bandwidth selection was applied. The convergence of this filtering method has been proved. Following [4], the selection is is based on the intuition that a good partition must be stable through scales. The bandwidth selection method is based on an iterative selection over the different feature subspaces. It allows a richer search of optimal analysis bandwidths than the non iterative method in [4]. The validity of our algorithm was shown on color image segmentation. Note that our algorithm has also been used for motion detection in [2], leading to very promising results. A direction of future research concerns the computation of covariance matrices. It would indeed be valuable to devise a new way of computing them that permits to capture the distribution near cluster's mode and the tails of a cluster.

References

[1] I. Abramson. On bandwidth variation in kernel estimates - a square root law. *The Annals of Statistics*, 10(4):1217–1223, 1982.

[2] A. Bugeau and P. Perez. Detection and segmentation of moving objects in highly dynamic scenes. *Proc. Conf. Comp. Vision Pattern Rec.*, 2007.

[3] Y. Cheng. Mean shift, mode seeking, and clustering. *IEEE Trans. Pattern Anal. Machine Intell.*, 17(8):790–799, 1995.

[4] D. Comaniciu. An algorithm for data-driven bandwidth selection. *IEEE Trans. Pattern Anal. Machine Intell.*, 25(2):281–288, 2003.

[5] D. Comaniciu and P. Meer. Mean shift: A robust approach toward feature space analysis. *IEEE Trans. Pattern Anal. Machine Intell.*, 24(5):603–619, 2002.

[6] D. Comaniciu and Peter Meer. Mean shift analysis and applications. In *Proc. Int. Conf. Computer Vision*, pages 1197–1203, 1999.

[7] D. Comaniciu, V. Ramesh, and P. Meer. The variable bandwidth mean shift and Data-Driven scale selection. *Proc. Int. Conf. Computer Vision*, 1, 2001.

[8] K. Fukunaga. *Introduction to statistical pattern recognition (2nd ed.)*. Academic Press Professional, Inc., San Diego, CA, USA, 1990.

[9] K. Fukunaga and L.D. Hostetler. The estimation of the gradient of a density function, with applications in pattern recognition. *IEEE Trans. Information Theory*, 21(1):32–40, 1975.

[10] P. Hall, T. Hui, and J. Marron. Improved variable window kernel estimates of probability densities. *The Annals of Statistics*, 23(1):1–10, 1995.

[11] A. K. Jain, M. N. Murty, and P. J. Flynn. Data clustering: a review. *ACM Comput. Surv.*, 31(3):264–323, 1999.

[12] W. Meisel L. Breiman and E. Purcell. Variable kernel estimates of multivariate densities. *Technometrics*, 19:135–144, 1977.

[13] Y. Leung, J. Zhang, and X. Zong-Ben. Clustering by scale-space filtering. *IEEE Trans. Pattern Anal. Machine Intell.*, 22(12):1396–1410, 2000.

[14] D. Loftsgaarden and C. Quesenberry. A nonparametric estimate of a multivariate density function. *Annals of Mathematical Statistics*, 36:1049–1051, 1965.

[15] Y.P. Mack and M. Rosenblatt. Multivariate k-nearest neighbor density estimates. *J. Multivariate Analysis*, 9:1–15, 1979.

[16] G. Milligan and M. Cooper. An examination of procedures for determining the number of clusters in a data set. *Psychometrika*, 50:159–179, 1985.

[17] B. Park and J.S. Marron. Comparison of data-driven bandwidth selectors. *J. of the American Statistical association*, 85(409):66–72, 1990.

[18] D. Pena and F.J. Prieto. Robust covariance matrix estimation and multivariate outlier detection. *Technometrics*, 43(3):286–310, 2001.

[19] D.W. Scott. Multivariate density estimation. *Wiley-Interscience*, 1992.

[20] S. Sheather and M. Jones. A reliable data-based bandwidth selection method for kernel density estimation. *J. Royal Statist. Soc.*, 53:683–690, 1991.

[21] B.W. Silverman. Density estimation for statistics and data analysis. *Chapman and Hall*, 1986.

[22] J.S. Simonoff. Smoothing methods in statistics. *Springer-Verlag*, 1996.

[23] G. Terrell and D. Scott. Variable kernel density estimation. *The Annals of Statistics*, pages 1236–1265, 1992.

[24] M. P. Wand and M. C. Jones. *Kernel Smoothing*. Chapman & Hall, London, U.K., 1995.

[25] N. Wang and Adrian E. Raftery. Nearest neighbor variance estimation (nnve): Robust covariance estimation via nearest neighbor cleaning. *J. of the American Statistical association*, 97(460):994–, December 2002.

Appendix

Proof of convergence of the pseudo balloon mean shift filtering

The balloon kernel density estimator is defined as:

$$\widehat{f}(\mathbf{x}) = \frac{c_k}{n} \sum_{i=1}^{n} \frac{1}{|\mathbf{H}(\mathbf{x})|^{1/2}} k(\|\mathbf{H}(\mathbf{x})^{-1/2}(\mathbf{x} - \mathbf{x}^{(i)})\|^2) \quad . \tag{A.1}$$

The proof of convergence of mean shift filtering using this estimator is closed to the one of the fixed bandwidth mean shift filtering [5]. We first show that $\widehat{f}$ is convergent for the trajectory points defined in algorithm 1, *i.e.* that $\widehat{f}(\mathbf{y}^{(j)})$ converges when j becomes large if $m(\mathbf{x})$ is defined as in (22). Since n is finite, $\widehat{f}$ is bounded : $0 < \widehat{f}(x) \leq \frac{c_k}{n|\mathbf{H}(\mathbf{x})|^{1/2}}$. It is then sufficient to show that $\widehat{f}$ is strictly increasing or decreasing. Since the bandwidth $\mathbf{H}(\mathbf{x})$ is constant for all trajectory points $\mathbf{y}^{(j)}$ associated to the

estimation point $\mathbf{x}$, we get:

$$\widehat{f}(\mathbf{y}^{(j+1)}) - \widehat{f}(\mathbf{y}^{(j)}) = \frac{c_k}{n|\mathbf{H}(\mathbf{x})|^{1/2}} \sum_{i=1}^{n} \left(k(\|\mathbf{H}(\mathbf{x})^{-1/2}(\mathbf{y}^{(j+1)} - \mathbf{x}^{(i)})\|^2) - k(\|\mathbf{H}(\mathbf{x})^{-1/2}(\mathbf{y}^{(j)} - \mathbf{x}^{(i)})\|^2) \right) . \tag{A.2}$$

The convexity of the profile k implies that:

$$\forall (x_1, x_2) \in [0, +\infty) \;\; k(x_2) \geq k(x_1) + k'(x_1)(x_2 - x_1) \;\; ,$$

and thus:

$$\widehat{f}(\mathbf{y}^{(j+1)}) - \widehat{f}(\mathbf{y}^{(j)}) \geq \frac{c_k}{n|\mathbf{H}(\mathbf{x})|^{1/2}} \sum_{i=1}^{n} k'(\|\mathbf{H}(\mathbf{x})^{-1/2}(\mathbf{y}^{(j)} - \mathbf{x}^{(i)})\|^2) \left(\|\mathbf{H}(\mathbf{x})^{-1/2}(\mathbf{y}^{(j+1)} - \mathbf{x}^{(i)})\|^2 - \|\mathbf{H}(\mathbf{x})^{-1/2}(\mathbf{y}^{(j)} - \mathbf{x}^{(i)})\|^2 \right). \tag{A.3}$$

We assume that $\mathbf{H}(\mathbf{x})^T = \mathbf{H}(\mathbf{x})$. Developing the last term and using the definition of the mean shift vector (equation 22) implies after some manipulations:

$$\widehat{f}(\mathbf{y}^{(j+1)}) - \widehat{f}(\mathbf{y}^{(j)}) \geq \frac{c_k}{n|\mathbf{H}(\mathbf{x})|^{1/2}} \sum_{i=1}^{n} k'(\|\mathbf{H}(\mathbf{x})^{-1/2}(\mathbf{y}^{(j)} - \mathbf{x}^{(i)})\|^2) \left(\|\mathbf{H}(\mathbf{x})^{-1/2}(\mathbf{y}^{(j+1)} - \mathbf{y}^{(j)})\|^2 \right). \tag{A.4}$$

Summing terms of previous equation for index $j, j+1, \ldots, j+m-1$, and introducing $M = \text{argmin}_{l \geq 0} k(\|\mathbf{H}(\mathbf{x})^{-1/2}(\mathbf{y}^{(l)} - \mathbf{x}^{(i)})\|^2$ results in:

$$\widehat{f}(\mathbf{y}^{(j+m)}) - \widehat{f}(\mathbf{y}^{(j)}) \geq \frac{c_k}{n|\mathbf{H}(\mathbf{x})|^{1/2}} M \|\mathbf{H}(\mathbf{x})^{-1/2}(\mathbf{y}^{(j+m)} - \mathbf{y}^{(j)})\|^2 \geq 0. \tag{A.5}$$

We have shown that the sequence $\{\widehat{f}(\mathbf{y}^{(j)})\}_{j=1,2\ldots}$ is strictly increasing, bounded, and thus convergent. It is also a Cauchy sequence. Inequality (A.5) implies that $\{\mathbf{y}^{(j)}\}_{j=1,2\ldots}$ is also a Cauchy sequence with respect to Mahalanobis norm, hence with respect to Euclidean norm (by vertue of norm equivalence in $\mathbb{R}^d$). This proves the convergence of trajectory points towards a local mode of $\widehat{f}$.

Unité de recherche INRIA Rennes
IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex (France)

Unité de recherche INRIA Futurs : Parc Club Orsay Université - ZAC des Vignes
4, rue Jacques Monod - 91893 ORSAY Cedex (France)
Unité de recherche INRIA Lorraine : LORIA, Technopôle de Nancy-Brabois - Campus scientifique
615, rue du Jardin Botanique - BP 101 - 54602 Villers-lès-Nancy Cedex (France)
Unité de recherche INRIA Rhône-Alpes : 655, avenue de l'Europe - 38334 Montbonnot Saint-Ismier (France)
Unité de recherche INRIA Rocquencourt : Domaine de Voluceau - Rocquencourt - BP 105 - 78153 Le Chesnay Cedex (France)
Unité de recherche INRIA Sophia Antipolis : 2004, route des Lucioles - BP 93 - 06902 Sophia Antipolis Cedex (France)

Éditeur
INRIA - Domaine de Voluceau - Rocquencourt, BP 105 - 78153 Le Chesnay Cedex (France)
http://www.inria.fr
ISSN 0249-6399

INRIA

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Bandwidth selection for kernel estimation in mixed multi-dimensional spaces

Aurélie Bugeau — Patrick Pérez

N° ????

September 2007

Thèmes COM et COG

Rapport de recherche

ISSN 0249-6399 ISRN INRIA/RR--????--FR+ENG

INRIA

FUTURS

INRIA

LORRAINE

INRIA

RHÔNE-ALPES

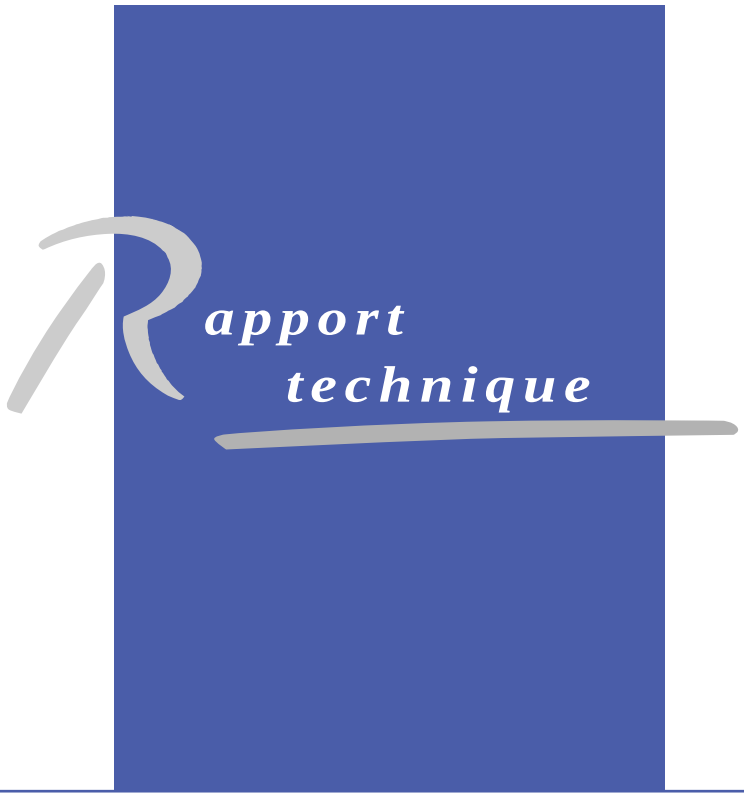
INRIA

ROCQUENCOURT

INRIA

SOPHIA ANTIPOLIS


INRIA


Rapport
technique